REMARKS

Applicant respectfully requests reconsideration of this application. Claims 1-30 are currently pending.

Claims 19, 21, and 23 have been amended. No claims have been cancelled. N0 claims have been added.

Therefore, claims 1-30 are now presented for examination.

Claim Rejection under 35 U.S.C. §103

McGreevy in view of Liao

The Examiner rejected claims 1-30 under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent Publication No. 2003/0078913 of McGreevy ("*McGreevy*") in view of U.S. Patent No. 7,054,315 of Liao ("*Liao*"). (*McGreevy* was subsequently issued as US Patent No. 6,823,333.)

Claim 1 is as follows:

1.   A method comprising:

grouping single fields of a multiple-field source into a plurality of multiple-field keys (MFKs) of a search target, each MFK of the search target having single fields that correspond to single fields in one of a plurality of multiple-field vectors (MFVs) of entries in a data structure;

generating a set of queries based, at least in part, on the MFKs, wherein each query includes one or more of the MFKs and wherein each query has a different MFK as a lead MFK;

using a query to determine whether the non-wildcard values in the MFVs of an entry match the non-wildcard values in corresponding MFKs of the search target; and

if no entry has non-wildcard values in the MFVs that match the corresponding non-wildcard values in the MFKs, then using the queries to

determine whether the entry has non-wildcard values in a MFV that match the non-wildcard values in a corresponding lead MFK, plus remaining MFVs that match corresponding remaining MFKs based on matching the non-wildcard values and wildcard values.

Claim Elements - Thus, claim 1 includes "grouping single fields of a multiple-field source into a plurality of multiple-field keys (MFKs) of a search target". The claim further provides that "each MFK of the search target having single fields that correspond to single fields in one of a plurality of multiple-field vectors (MFVs) of entries in a data structure". Claim 1 further provides for "generating a set of queries based, at least in part, on the MFKs, wherein each query includes one or more of the MFKs and wherein each query has a different MFK as a lead MFK".

The claim also includes further claim elements regarding a search, providing for using a query to determine whether the non-wildcard values in MFVs of an entry match the non-wildcard values in corresponding MFKs of the search target, and, if no entry has non-wildcard values in the MFVs that match the corresponding non-wildcard values in the MFKs, then using the queries to determine whether the entry has non-wildcard values in a MFV that match the non-wildcard values in a corresponding lead MFK, plus remaining MFVs that match corresponding remaining MFKs based on matching the non-wildcard values and wildcard values.

Argument Raised by Applicant – The Response to Arguments of the current Office Action indicates the following:

> Applicant's arguments filed on August 13, 2008, with respect to claim 1 on page 11 and through page 18 of the remarks, have been fully

considered. Basically the applicant argued that Liao does not teach or suggest "single fields of a multiple-field source". The Examiner has introduced a new reference which [teaches] or suggests "single fields of multiple-field source". See the above rejections for the relevant interpretation and citations founds found in McGreevy et al., disclosing the limitation.

To correct this assertion, the argument presented by the Applicant is NOT that the reference fails to teach or suggest "single fields of a multiple-field source".

What the Applicant is arguing is that, rather than <u>grouping together fields of a multiple field source into a search target having multiple-field keys and generating queries using these multiple field keys</u>, *Liao* is concerned with grouping together candidate bit patterns.

With regard to the newly cited *McGreevy* reference, as described more fully below, there is general discussion regarding forming queries from keyword searches. However, it is again submitted that there is no showing of grouping together fields of a multiple field source into a search target having multiple-field keys and generating the queries using these multiple field keys as is provided in claim 1.

McGreevy Reference – The newly cited *McGreevy* patent application regards keyterm searching, and specifically regards a system and method of "analyzing data in context". (*McGreevy*, ¶0001) As described in the reference, a query can include one or more keyterms, and a "gleaning" model of the query is created. The gleaning model is compared to relational models of subsets of the database, and the identifiers of the relevant subsets are then output. (*McGreevy*, abstract)

The *McGreevy* reference is summarized as:

[0014] In accordance with one aspect of the present invention, a keyterm search is a method of searching a database for subsets of the database that are relevant to an input query. First, a number of relational models of subsets of a database are provided. A query is then input. The query can include one or more keyterms. Next, a gleaning model of the query is created. The gleaning model of the query is then compared to each one of the relational models of subsets of the database. The identifiers of the relevant subsets are then output.

(*McGreevy*, ¶0014) Thus, the reference regards searching a database for "subsets of the database that are relevant to an input query". The steps to accomplish this are:

(1) Providing a number of "relational models of subsets of a database";

(2) Creating a "gleaning model" of the query;

(3) Comparing the gleaning model to each of the relational models of subsets of the database; and

(4) Outputting identifiers of the relevant subsets.

Input queries obviously can include multiple elements. "[T]he input query can include a single term or multiple terms." (*McGreevy*, ¶0108, lines 1 and 2) As further indicated by the reference:

[0165] Keyterm search will take any number of keyterms as queries, as in the above examples, but each term is treated individually. A search on the keyterms "frequency congestion" will return narratives that contain either one or both of these keyterms and their corresponding contexts. There is no guarantee, however, that both of the keyterms will appear in the top-ranked narratives because the search treats each query term as an independent item.

(*McGreevy*, ¶0165) This indicates that a keyterm search will take any number of

keyterms as queries, and that each term is treated individually. However, this does not

suggest the generation of queries as provided in claim 1.

As indicated by *McGreevy*, keyterm searching is the following:

[0101] Keyterm search is a method of retrieving from a database a
number of subsets of the database that are most relevant to a criterion
model derived from one or more keyterms. The retrieved subsets can also
be ranked according to their corresponding relevance to the criterion
model. One embodiment of a keyterm search is a method of searching a
database. First, several relational models are provided. Each one of the
relational models includes one relational model of at least one subset of
the database. Next, a query is input. A criterion model is then created. The
criterion model is a relational model that is based on the query. The
criterion model is then compared to each one of the relational models of
subsets. The identifiers of the subsets relevant to the query are then output.

(*McGreevy*, ¶0101) Thus, as described, a criterion model is created, where the criterion

model is a relational model that is based on the query. However, the question remains

what query is being presented. As further described in the next paragraph of the

reference with regard to figures 7-10 of the reference: "A query is input in block 704 for

comparing to the relational models of subsets of the database. The query can include one

term or multiple terms. Next, the query is expanded and modeled to create a criterion

model in block 708, as will be more fully described below. The criterion model is then

compared to each one of the relational models of subsets of the database in block 710 that

is also described in more detail below. The identifiers of the relevant subsets are then

output in block 712."

Further, the reference discusses use of a query model:

[0103] As an alternative form of input to the keyterm search process, the input query can consist of a query model. A query model can provide detailed control of the relevance criteria embodied in an input query. <u>As a further alternative, the input query can consist of a selected portion of a previously output query model. One alternative method of selecting a portion of an output query model includes selecting a number of relations whose term pairs contain any of a selected group of terms. Another alternative method of selecting a portion of an output query model includes selecting a number of relations having selected metrics greater than a selected threshold value. As another alternative, the input query model can be a model of a subset of a database. As another alternative, the input query model can be a model of a subset of a database having relational metrics that have been multiplied by one or more of a collection of scale factors. As a further alternative, the input query model can be created by manually creating term pairs and corresponding metric values.</u> When a query model is used as an input query, the process of expanding the query and creating a relational model of the query shown in block 708 includes passing the input query model to the comparing process shown in block 710.

(*McGreevy*, ¶0103) Thus, the reference discusses generation of an input query. However the suggestion provided here regards the selection of a previously output query model, or a model of a subset of a database having relational metrics multiplied by scale factors, or the creation of a input query model manually using term pairs and metric values. There is no discussion of the generation of a query as provided in claim 1.

Further, a query can be expanded to include related concepts, which is what is referred to as creating a gleaning model. (*McGreevy*, ¶¶0111-0112) However, the result

here is additional terms to a query, which does not teach or suggest the elements of claim 1.

The *McGreevy* reference further describes the relational model used in searches. The process for producing a relational model is shown in Figure 1, and a process for combining relational models of databases is shown in Figure 2. Figure 6A of the reference illustrates a relational model, in which a diagram of the relationship between elements in the databases is provided. Figure 10 then shows a process for comparing a relational model of a query to relational models of subsets of the database.

Thus, the *McGreevy* reference provides a particular process in which relational models are used to provide comparisons between a query and a database. However, this does not teach or suggest the elements of claim 1. Even if it is assumed for the sake of argument that the use of a relational model as described in *McGreevy* provides for grouping single fields of a multiple-field source into a plurality of multiple-field keys (MFKs) of a search target, and that each MFK of the search target has single fields that correspond to single fields in one of a plurality of multiple-field vectors (MFVs) of entries in a data structure, then the reference still would contain no teaching or suggestion of generating a set of queries based, at least in part, on the MFKs, wherein each query includes one or more of the MFKs and wherein each query has a different MFK as a lead MFK.

As the *McGreevy* reference does not teach or suggest the queries provided by claim 1, the reference further does not provide for using such a query to determine whether the non-wildcard values in MFVs of an entry match the non-wildcard values in corresponding MFKs of the search target, and, if no entry has non-wildcard values in the

MFVs that match the corresponding non-wildcard values in the MFKs, then using the queries to determine whether the entry has non-wildcard values in a MFV that match the non-wildcard values in a corresponding lead MFK, plus remaining MFVs that match corresponding remaining MFKs based on matching the non-wildcard values and wildcard values.

Liao Reference – As argued in the previous response, *Liao* regards masked matching, and is intended to reduce the search space that is processed by mask matching methods. The elements that are being grouped in *Liao* are possible related bit patterns. "The search space is reduced by grouping the candidate bit patterns into groups and subgroups that have internal bit agreement between the members. By only applying the mask matching methods to a select number of groups selected by their bit agreement with the target bit pattern, the computation time and memory requirement of the mask matching method is reduced." (*Liao*, Abstract) While searching efficiency is intended by *Liao*, the reference is addressing this general goal in a different manner.

The Office Action apparently is now citing to elements 170 (PS0) and 180 (PS1) of Figure 5 as MFKs. However, it is again submitted that what is represented in Figure 5 does not relate to multiple fields of any kind, but rather multiple possible search candidates. As described in the prior response, a Patricia tree is a data structure used in search database, the tree being a binary tree with each internal node having two branches and bit index value. (*Liao*, col. 7, lines 10-14) Figure 1 illustrates a Patricia tree using rule set shown in a table (*Liao*, col. 7, lines 45-52) (where a rule set is a collection of filter rules). The elements of the table are <u>not</u> fields of a source, but rather various candidates for searching. Stated in another way, these are possible bit patterns to match,

representing multiple possible values of the six-bit field that is being used in this example. *Liao* then intends to use the concept of the Patricia tree to reduce the search space needed for a match, which is needed because matching the rule set with a wild card requires expanding to cover the wild card space. (*Liao*, col. 8, lines 14-22)

With this background, Figure 4 describes the generation of elements 130, 140, 150, and 160. From the text of *Liao*, it can be seen that is a certain set of candidate bit patterns shown in Table 1. (*Liao*, col. 10, lines 9-20) Table 1 does contain a rule set that has wild card or "don't care" (represented as 'x') values. Given this, the tree 100 of Figure 4 is generated, which contains group1 (having the candidates with '1' in the initial bit), group0 ('0' in the initial bit), and groupx ('x' – the wild card bit – in the initial bit). These then are divided into subgroups, boxes 110 and 120 using the third and fourth bit positions. Ultimately the subgroups represented in boxes 130, 140, 150, and 160 are produced, with the elements in each group having common bit patterns.

From this result, the process proceeds with packaging the subgroups into "physical sets" in Figure 5 – PS0 and PS1 – where the elements that are common to the subgroups are only represented once in the physical set. Once this is done, there is used in a determination whether the target bit pattern matches a candidate bit pattern. This can be seen in Figure 6, in which a search proceeds from the '0' index position ('1' in this example) to the '4' index position (value of '0'), which then results in calling a particular index (2), which denotes a particular physical set (PS1). This is retrieved, and the matching pattern is applied to the physical set using the target bit pattern. (*Liao*, col. 12, lines 49-65)

Thus, it is again submitted that what is being done is *Liao* is very different than what is described in claim 1. Rather than grouping together fields of a multiple field source into a search target having multiple-field keys and generating queries using these multiple field keys, *Liao* is concerned with grouping together candidate bit patterns.

To describe this diagrammatically, claim 1 regards a multiple field source (such as Figure 5A of the current application). These are not <u>values</u> for a particular field (as in *Liao*), but rather are <u>single fields</u> of a source that are grouped together into the MFKs. The search target has these grouped MFKs. The queries are generated based, at least in part, on the MFKs, with each query having a different MFK (a different grouping of fields) as a lead MFK. In contrast, what *Liao* is concerned about is the <u>possible values for any particular field</u> in a search and how this affects searching. These values are grouped together, and the grouped values are used in searching, as shown in, for example, Figure 6 of *Liao*.

Thus, neither *McGreevy* nor *Liao* contains the elements of claim 1 as described above, and claim 1 is patentable over *McGreevy* and *Liao*. It is submitted that the arguments presented above with regard to claim 1 also apply to independent claims 10, 19, and 28, and such claims are also allowable.

The remaining claims, while having other differences with the cited references are dependent claims, and thus are allowable as being dependent on the allowable base claims.

Conclusion

Applicant respectfully submits that the rejections have been overcome by the amendment and remark, and that the claims as amended are now in condition for

allowance. Accordingly, Applicant respectfully requests the rejections be withdrawn and

the claims as amended be allowed.

Invitation for a Telephone Interview

The Examiner is requested to call the undersigned at (503) 439-8778 if there remains any issue with allowance of the case.

Request for an Extension of Time if Needed

The Applicant respectfully petitions for an extension of time to respond to the outstanding Office Action pursuant to 37 C.F.R. § 1.136(a) should one be needed. Please charge any fee to our Deposit Account No. 02-2666.

Charge our Deposit Account

Please charge any shortage to our Deposit Account No. 02-2666.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Date: July 28, 2008   /Mark C. Van Ness/
                         Mark C. Van Ness
                         Reg. No. 39,865

1279 Oakmead Parkway
Sunnyvale, CA 94085-4040
(503) 439-8778